```
//Program 4
//CIS 199-02
//GRADING ID: B5992
//DUE: Tuesday, 4/24/2017
//This program collects information about a package and then tells you the cost to ship the package from the
//specified origin zip to the specified destination zip in a list box. It also contains features such as the
//details button that tells you all the details of the package as well as the send to uofl and send from uofl
//buttons that change the origin or dest zip and recalculate the price.
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Program_4
{
    public partial class program4 : Form
    {
        //List that holds packages.
        private List<GroundPackage> packageList = new List<GroundPackage>();

        //Precondition: None
        //Postcondition: Form is made, ready to load.
        public program4()
        {
            InitializeComponent();
        }

        //Precondition: None
        //Postcondition: Form data is gathered and parsed. If all parses then method returns isValid as true.
        private bool ValidatePackageData(out int oZip, out int dZip, out double length, out double height, out double width, out double weight)
        {
            oZip = 0;
            dZip = 0;
            length = 0;
            height = 0;
            width = 0;
            weight = 0;

            bool isValid = false; //To hold validation variable.

            if(int.TryParse(originTxt.Text, out oZip))
            {
                if (int.TryParse(destinationTxt.Text, out dZip))
                {
                    if (double.TryParse(lengthTxt.Text, out length))
                    {
                        if (double.TryParse(heightTxt.Text, out height))
                        {
                            if (double.TryParse(widthTxt.Text, out width))
                            {
                                if (double.TryParse(weightTxt.Text, out weight))
                                {
                                    isValid = true;
                                }
                                else
                                    MessageBox.Show("Please enter a valid weight.");
                            }
                            else
                                MessageBox.Show("Please enter a valid width.");
                        }
                        else
                            MessageBox.Show("Please enter a valid height.");
                    }
                    else
                        MessageBox.Show("Please enter a valid length");
                }
                else
                    MessageBox.Show("Please enter a valid destination zip code.");
            }
            else
                MessageBox.Show("Please enter a valid origin zip code.");

            return isValid;
        }

        //Precondition: addButton is clicked.
        //Postcondition: If data is valid a new package is added to the list box.
        private void addButton_Click(object sender, EventArgs e)
        {
            GroundPackage myPackage; //Holds new package from valid data
            int oZip; //To hold entered origin zip.
            int dZip; //To hold entered destination zip.
            double height; //To hold entered height.
            double length; //To hold entered length.
            double width; //To hold entered width.
            double weight; //To hold entered weight.

            if (ValidatePackageData(out oZip, out dZip, out length, out height, out width, out weight))
            {
                //Create new package
                myPackage = new GroundPackage(oZip, dZip, length, height, width, weight);

                packageList.Add(myPackage);

                outputListBox.Items.Add(myPackage.CalcCost().ToString( "c"));

                originTxt.Clear();
                destinationTxt.Clear();
                lengthTxt.Clear();
                heightTxt.Clear();
                widthTxt.Clear();
                weightTxt.Clear();

                originTxt.Focus();

            }
        }

        //Precondition: Details button is clicked.
        //Postcondition: Displays a message box with package info as long as index is valid.
        private void detailsButton_Click(object sender, EventArgs e)
        {
            int index;

            index = outputListBox.SelectedIndex;

            if (index >= 0)
            {
                MessageBox.Show(packageList[index].ToString());
            }
            else
                MessageBox.Show("Please select a package.");
        }

        //Precondition: sendToButton is clicked.
        //Postcondition: Displays message that destination zip was updated and updates destination zip to 40292.
        private void sendToButton_Click(object sender, EventArgs e)
        {
            int index;

            index = outputListBox.SelectedIndex;

            if (index >= 0)
            {
                packageList[index].DestinationZip = 40292;
                outputListBox.Items[index] = packageList[index].CalcCost
                  ().ToString("c");
                MessageBox.Show("Destination zip code has been updated!");
            }
            else
                MessageBox.Show("Please select a package.");
        }

        //Precondition: sendFromButton is clicked.
        //Postcondition: Displays a message that origin zip has been updated
          and origin zip is updated to 40292.
        private void sendFromButton_Click(object sender, EventArgs e)
        {
            int index;

            index = outputListBox.SelectedIndex;
            if (index >= 0)
            {
                packageList[index].OriginZip = 40292;
                outputListBox.Items[index] = packageList[index].CalcCost
                  ().ToString("c");
                MessageBox.Show("Origin zip code has been updated!");
            }
            else
                MessageBox.Show("Please select a package.");
        }
    }
}
```